PUBLIC KEY ENCRYPTION METHOD AND COMMUNICATION SYSTEM USING PUBLIC KEY CRYPTOSYSTEM

BACKGROUND OF THE INVENTION

The present invention relates to a cipher communication method and a key sharing method that uses public key cryptosystem.

Various public key encryption schemes have been so far proposed. Of these, a method described in document 1, "R.L. Rivest, A.Shamir, L.Adleman: A method for obtaining digital signatures and public-key cryptosystems, Commun. of the ACM, Vol. 21, No. 2, pp. 120-126, 1978" is the most famous and most practically used public key cryptosystem. Additionally, methods using elliptic curves, described in document 2 "V.S.Miller: Use of Elliptic Curves in Cryptography, Proc. of Crypto '85, LNCS218, Springer-Verlag, pp. 417-426 (1985)", and document 3 "N.Koblitz: Elliptic Curve Cryptosystems, Math. Comp., 48, 177, pp. 203-209 (1987)", etc., are known as efficient public key cryptosystems.

Known encryption methods provably secure against chosen plaintext attacks include those described in: document 4 "M.O.Rabin: Digital Signatures and Public-Key Encryptions as Intractable as Factorization, MIT, Technical Report, MIT/LCS/TR-212 (1979); document 5 "T.ElGamal: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE Trans. On Information Theory, IT-31, 4, pp. 469-472 (1985)"; document 6 "S.Goldwasser and S.Micali: Probabilistic Encryption, JCSS, 28, 2, pp. 270-299 (1984)"; document 7 "M.Blum and S.Goldwasser: An Efficient probabilistic public-key encryption scheme which hides all partial information, Proc. of Crypto '84, LNCS196, Springer-Verlag, pp. 289-299 (1985); document 8 "S. Goldwasser and M.Bellare: Lecture Notes on Cryptography, http:/www-cse.ucsd.edu/users/mihir/(1997)"; and document 9 "T. Okamoto and S. Uchiyama: A New Public-Key Cryptosystem as Secure as

Factoring, Proc. of Eurocrypt '98, LNCS1403, Springer Verlag, pp. 308-318 (1998)". Known encryption methods provably secure against chosen ciphertext attacks include those described in: document 10 "D.Dolve, C.Dwork and M.Naor: Non-malleable cryptography, In 23rd Annual ACM Symposium On Theory of Computing, pp. 542-552 (1991)"; document 11 "M.Naor and M.Yung: Public-key cryptosystems provably secure against chosen ciphertext attacks, Proc. of STOC, ACM Press, pp. 427-437 (1990)"; document 12 "M.Bellare and P.Rogaway, Optimal Asymmetric Encryption How to Encrypt with RSA, Proc. of Eurocrypt '94, LNCS950, Springer Verlag, pp. 92-111 (1994)"; and document 13 "R.Cramer and V.Shoup: A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack, Proc. of Crypto98, LNCS1462, Springer-Verlag, pp. 13-25 (1998)".

In document 14 "M.Bellare, A.Desai, D.Pointcheval and P.Rogaway.: Relations Among Nations of Security for Public-Key Encryption Schemes, Proc. of Crypto '98, LNCS1462, Springer Verlag, pp. 26-45 (1998)", there is shown the equivalence between IND-CCA2 (indistinguishable against adaptive chosen ciphertext attacks) and NM-CCA2 (non-malleable against adaptive chosen ciphertext attacks). Presently, public key cryptosystem satisfying this condition is considered to be the most secure.

SUMMARY OF THE INVENTION

The present invention provides a public key encryption method that is provably secure and excellent in the efficiency of encryption and decryption processing.

The present invention first provides a public key encryption method that is provably OW-CPA (unidirectional for chosen plaintext attacks), under the assumption that the prime factorization problem is computationally intractable. The present invention also provides a public key encryption method that is provably IND-CCA2 (or NM-CCA2) which is based on this method.

These encryption methods are smaller in the number of modular multiplications required in encryption and decryption processing than conventional methods, enabling high-speed processing.

Also, the present invention provides an encryption method and a decryption method using public key cryptosystem which produce a small amount of computational load in encrypting send data and decrypting encrypted data and enables high-speed processing for devices with limited computational capability such as portable information processing equipment, a key distribution method and a key sharing method using these methods, and programs, devices, or systems that implement the methods.

The present invention is performed as follows.

(1) As $n=p^dq$ (d is an odd number satisfying d>1), for the bit length k of pq, a small plaintext space is selected so as to be an open set $(0, 2^{k-2})$.

(2) On a residue group modulo a composite number (a number consisting of products of plural mutually different prime integers), there are four or more square roots, and by putting the solutions of these square roots to good use, n can be factorized into prime integers. Taking advantage of this fact, the public key encryption method of the present invention builds a procedure for encryption and decryption so as to be provably secure for chosen plaintext attacks(OW-CPA), under the assumption that the problem of prime factorization is intractable.

(3) For a public key encryption method by the above (1) and (2), the transformation method described in the document 12 is executed for transformation into a method having more powerful security, under the assumption that (ideal) random functions are publicized.

As one concrete method,

[Key generation]

a secret key (private key) (p,q,β) satisfying

- $p,\ q$: prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$

is generated, and a public key (n,k,k$_0$,k$_1$,α,G,H) satisfying

- $n = p^d q$ $(d > 1$ is odd)
- k, k_0, k_1 : k is a binary length of pq, and k_0, k_1 are positive integers with $k > k_0 - k_1 - 2$.
- $\alpha \in \mathbb{Z}$
- $G : \{0,1\}^{k_0} \to \{0,1\}^{k-k_0-2}$
- $H : \{0,1\}^{k-k_0-2} \to \{0,1\}^{k_0}$

is generated.

[Encryption]

A sender device computes

$$x = (m0^{k_1} \odot G(r)) || (r \odot H(m0^{k_1} \odot G(r)))$$

where a circled dot denotes "exclusive OR"

for plaintext m (m∈$\{0,1\}^l$,l=k-k$_0$-k$_1$-2) and a random number r(r∈$\{0,1\}^{k0}$},

$$C = x^{2n\alpha} \bmod n$$

further computes

and further computes Jacobi's symbol a=(x/n), and sends ciphertext (C,a) to the receiver device.

[Decryption]

$$\begin{aligned} x_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ x_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

The receiver device computes

from the ciphertext (C,a), using a receiver's secret key (private key) (p,q,β),

and computes y that satisfies (y/n)=a and 0<y<2^{k-2} of $\phi(x_{1,p}, x_{1,q})$,

$\phi(-x_{1,p},x_{1,q})$, $\phi(x_{1,p},-x_{1,q})$, and $\phi(-x_{1,p},-x_{1,q})$, where ϕ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem. Furthermore,
when

$$y = s||t \quad (s \in \{0,1\}^{k-k_0-2},\ t \in \{0,1\}^{k_0})$$

the receiver device computes

$$z = G(H(s) \odot t) \odot s,$$

$$m = \begin{cases} [z]^l & \text{if } [z]_{k_1} = 0^{k_1} \\ \text{“reject”} & \text{otherwise} \end{cases}$$

and decrypts the plaintext m by

where $[a]^k$ and $[a]_k$ denote first k-bits and last k-bits of a, respectively.

These and other benefits are described throughout the present specification. A further understanding of the nature and advantages of the invention may be realized by reference to the remaining portions of the specification and the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Preferred embodiments of the present invention will be described in detail based on the followings, wherein:

FIG. 1 is a diagram showing the system configuration of embodiments of the present invention;

FIG. 2 is a diagram showing the internal configuration of a sender device in embodiments of the present invention;

FIG. 3 is a diagram showing the internal configuration of a receiver device in embodiments of the present invention;

FIG. 4 is a diagram showing the internal configuration of a storage medium with a computing function in embodiments of the present invention;

FIG. 5 is a diagram showing the outline of a first embodiment example;

FIG. 6 is a diagram showing the outline of a sixth embodiment example;

FIG. 7 is a diagram showing the outline of a seventh embodiment example;

FIG. 8 is a diagram showing the outline of a ninth embodiment example;

FIG. 9 is a diagram showing the outline of an eleventh embodiment example; and

FIG. 10 shows comparisons between the method of an eleventh embodiment example ($\alpha=\beta=1$) and a typical practical public key encryption method in efficiency (the number of modular products) and security.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Hereinafter, embodiment examples of the present invention will be described with reference to the accompanying drawings.

As shown in FIG. 1, a system of embodiment examples of the present invention includes a sender device 100 and a receiver device 200. Further, the sender device 100 and the receiver device are connected over a communication line 300.

As shown in FIG. 2, the sender device includes a random number generating unit 101, an exponentiation unit 102, an operation unit 103, a modulo calculation unit 104, a memory 105, a communication device 106, and an input device 107.

As shown in FIG. 3, the receiver device 200 includes a key generating unit 201, an exponentiation unit 202, a modulo calculation unit 203, an operation unit 204, a memory 205, and a communication device 206.

As shown in FIG. 4, a storage medium with a computing function 400 includes an exponentiation unit 401, a modulo calculation unit 402, an operation unit 403, a memory 404, an output device 405, a plaintext creating unit 406, and a random number generating unit 407.

Any of the sender device 100, the receiver device 200, and the storage medium with a computing function 400 can be constructed using a computer having a CPU and a memory. Any of the random number generating unit, the key generating unit, the power computing unit, the modulo calculation unit, the plaintext creating unit, and the random number generating unit may be constructed with dedicated hardware or as a program running on an operation unit (CPU). The programs are embodied on computer-readable media such as portable storage media and communication media on a communication line, and are stored in a computer memory through the media.

(First embodiment example)

In the present embodiment example, a message sender A sends send data m to a receiver B over cipher communications.

FIG. 1 shows the system configuration of the present embodiment example. FIG. 5 outlines this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β) satisfying

- p, q : prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,α) (k denotes the bit length of pq) satisfying

- $n = p^d q$ $\quad$ ($d > 1$ is odd)
- k : binary length of pq
- $\alpha \in \mathbb{Z}$

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Encryption and decryption processing

(1) The sender A computes

$$C = m^{2n\alpha} \bmod n$$

by using the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 for plaintext m ($0 < m < 2^{K-2}$).

Furthermore, the sender A obtains the above public information from the receiver B and computes Jacobi's symbol a=(m/n) using the operation unit 103 within the sender device 100 (the definition and computation method of the Jacobi's symbol are described in, e.g., Teiji Takagi, "Elementary Number System", Iwanami Shoten, Publishers).

Furthermore, the sender A sends ciphertext (C,a) to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

(2) The receiver B computes

from the ciphertext (C,a), using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and regards as the plaintext m any of $\phi(m_{1,p},m_{1,q})$, $\phi(-m_{1,p},m_{1,q})$, $\phi(m_{1,p},-m_{1,q})$, and $\phi(-m_{1,p},-m_{1,q})$ that satisfies $(x/n)=a$ and $0<x<2^{k-2}$, where φ denotes ring isomorphism mapping from $Z/(p)\times Z/(q)$ to $Z/(pq)$ by the Chinese remainder theorem.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions $p=2p'+1$ and $q=2q'+1$, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system. Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

According to a method in the present embodiment example, for example, when d=3, it can be proved that perfect decryption is impossible, under the assumption that the problem of prime factorization of n is intractable. Namely, if an algorithm for solving the problem of prime factorization of n is available, the algorithm could be used to form an algorithm for perfect decryption.

(Second embodiment example)

In this embodiment example, a, which is part of ciphertext in the first embodiment example, is used as a public key.

FIG. 1 shows the system configuration of this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β) satisfying

- $p,\ q$: prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,α,a) (k denotes the bit length of pq)

- $n = p^d q \qquad (d > 1 \text{ is odd})$
- k : binary length of pq
- $\alpha \in \mathbb{Z}$
- $a \in \{-1, 1\}$

satisfying

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Encryption and decryption processing

(1) The sender A computes

$$C = m^{2n\alpha} \bmod n$$

by using the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 for plaintext m ($0 < m < 2^{K-2}$) satisfying a=(m/n).

Furthermore, the sender A sends ciphertext C to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

(2) The receiver B computes

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

from the ciphertext (C,a), using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and regards as the plaintext m any of $\phi(m_{1,p},m_{1,q})$, $\phi(-m_{1,p},m_{1,q})$, $\phi(m_{1,p},-m_{1,q})$, and $\phi(-m_{1,p},-m_{1,q})$ that satisfies (x/n)=a and $0<x<2^{k-2}$, where ϕ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions p=2p'+1 and q=2q'+1, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system. Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

(Third embodiment example)

In this embodiment example, a description will be made of a method of creating plaintext m so as to include check information for checking whether message text to be sent to a receiver from a sender has been correctly decrypted. It can be proved that the public key encryption method in the first and second embodiment examples is unidirectional for chosen plaintext attacks, but it is not secure against chosen ciphertext attacks. Accordingly, message text to be sent to a receiver from a sender is transformed into plaintext m whose contents are provided with predetermined redundancy, the plaintext m is encrypted by the method described in the first embodiment example (or second embodiment example), and the receiver decrypts the plaintext m by the method of the first embodiment example (or second embodiment example) and checks the predetermined redundancy (if the predetermined redundancy is not provided, it is considered that decryption was not performed correctly).

As another method, message text to be sent to a receiver from a sender is transformed into plaintext m whose contents are provided with a predetermined, meaningful message, the plaintext m is encrypted by the method described in the first embodiment example (or second embodiment example), and the receiver decrypts the plaintext m by the method of the first embodiment example (or second embodiment example) and checks the contents of the predetermined, meaningful message (if the contents of the predetermined, meaningful message do not match, it is considered that decryption was not performed correctly).

These methods provide the public key encryption method of the first and second embodiment examples with some degree of security against chosen ciphertext attacks (a method of proving security against chosen ciphertext attacks will be described in embodiment examples).

(Fourth embodiment example)

In this embodiment example, a description will be made of a key sharing method for sharing an identical value between a sender and a receiver, using public information generated by the receiver.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β)

- $p,\ q$: prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\operatorname{lcm}(p-1, q-1)}$

satisfying

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,α,f) (k denotes the bit length of pq) satisfying

- $n = p^d q \qquad (d > 1 \text{ is odd})$
- k : binary length of pq
- $\alpha \in \mathbb{Z}$
- f : one-way function

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Key distribution processing

(1) The sender A computes

$$C = m^{2n\alpha} \bmod n$$

by using the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 for plaintext m $(0 < m < 2^{K-2})$.

Furthermore, the sender A obtains the above public information from a third party or the receiver B and computes Jacobi's symbol a=(m/n) using the operation unit 103.

Furthermore, the sender sends ciphertext (C,a) to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

Also, the sender computes shared key K=f(m) using the operation unit 103 and the modulo calculation unit 104 within the sender device 100 from a unidirectional function f, which is public information.

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

(2) The receiver B computes

from the ciphertext (C,a), using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and regards as the plaintext m any of $\phi(m_{1,p},m_{1,q})$, $\phi(-m_{1,p},m_{1,q})$, $\phi(m_{1,p},-m_{1,q})$, and $\phi(-m_{1,p},-m_{1,q})$ that satisfies $(x/n)=a$ and $0<x<2^{k-2}$, where ϕ denotes ring isomorphism mapping from $Z/(p)\times Z/(q)$ to $Z/(pq)$ by the Chinese remainder theorem. Furthermore, the receiver B computes shared key $K=f(m)$ using the operation unit 204, from the unidirectional function f, which is public information.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions $p=2p'+1$ and $q=2q'+1$, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system. Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

(Fifth embodiment example)

In this embodiment example, a, which is part of ciphertext in the first embodiment example, is used as a public key.

FIG. 1 shows the system configuration of this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β) satisfying

- $p,\ q$: prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\operatorname{lcm}(p-1, q-1)}$

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,α,a,f) (k denotes the bit length of pq)

- $n = p^d q$ $(d > 1$ is odd$)$
- k : binary length of pq
- $\alpha \in \mathbb{Z}$
- $a \in \{-1, 1\}$
- f : one-way function

satisfying

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Key distribution processing

(1) The sender A computes

$$C = m^{2n\alpha} \bmod n$$

by using the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 for plaintext m ($0 < m < 2^{K-2}$) satisfying a=(m/n) (a=(m/n) denotes Jacobi's symbol).

Furthermore, the sender sends ciphertext C to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

Also, the sender computes shared key K=f(m) using the operation unit 103 and the modulo calculation unit 104 from the unidirectional function f, which is public information.

(2) The receiver B computes

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

from the ciphertext C, using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and regards as the plaintext m any of $\phi(m_{1,p},m_{1,q})$, $\phi(-m_{1,p},m_{1,q})$, $\phi(m_{1,p},-m_{1,q})$, and $\phi(-m_{1,p},-m_{1,q})$ that satisfies (x/n)=a and $0<x<2^{k-2}$, where ϕ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem. Furthermore, the receiver B computes shared key K=f(m) using the operation unit 204, from the unidirectional function f, which is public information.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions p=2p'+1 and q=2q'+1, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system. Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

(Sixth embodiment example)

In this embodiment example, a description will be made of how the storage medium with a computing function 400 which has poor computation capability such as an IC card computes ciphertext C, using the sender device 100 having high computation capability in the first to fifth embodiment examples. FIG. 6 outlines this embodiment example.

The storage medium with a computing function 400 generates plaintext m ($0 < m < 2^{K-2}$), using the plaintext creating unit 406.

Furthermore, the storage medium with a computing function 400

$$C_1 = m^{2\alpha} \bmod n$$

computes

using the power computing unit 401 and the modulo calculation unit 402 from the public keys α and n, and outputs it to the input device 107 of the sender device 100 from the output device 405.

The sender device 100 uses the power computing unit 202 and the

$$C = {C_1}^n \bmod n$$

modulo calculation unit 203 to compute ciphertext C by

.

(Seventh embodiment example)

In this embodiment example, by the transformation method described in the document 12 (described in "Prior Art"), the public key encryption method of the first embodiment example is transformed into a public key encryption method provably secure against adaptive chosen ciphertext attacks.

FIG. 1 shows the system configuration of this embodiment example. FIG. 7 outlines this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β) satisfying

- p, q : prime integers, $p \equiv 3 \pmod 4$, $q \equiv 3 \pmod 4$
- $\beta \in \mathbb{Z}$, $\alpha\beta \equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,k₀,k₁,α,G,H) (k denotes the bit length of pq) satisfying

- $n = p^d q$ $\quad$ ($d > 1$ is odd)
- k, k_0, k_1 : k is a binary length of pq, and k_0, k_1 are positive integers with $k > k_0 - k_1 - 2$.
- $\alpha \in \mathbb{Z}$
- $G : \{0,1\}^{k_0} \to \{0,1\}^{k-k_0-2}$
- $H : \{0,1\}^{k-k_0-2} \to \{0,1\}^{k_0}$

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Encryption and decryption processing

(1) The sender A selects a random number r($r \in \{0,1\}^{k0}$} for plaintext m ($m \in \{0,1\}^l$, $l=k-k_0-k_1-2$) by using the random number generating unit 101, uses the operation unit 103 within the sender device 100 to compute

$$x = (m0^{k_1} \odot G(r)) || (r \odot H(m0^{k_1} \odot G(r)))$$

and further uses the operation unit 103, the power computing unit 102,

$$C = x^{2n\alpha} \bmod n$$

and the modulo calculation unit 104 to compute

Furthermore, the sender A obtains the above public information from a third party or the receiver B and computes Jacobi's symbol a=(x/n) using the operation unit 103.

Furthermore, the sender A sends ciphertext (C,a) to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

(2) The receiver B computes

$$
\begin{aligned}
x_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\
x_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q
\end{aligned}
$$

from the ciphertext (C,a), using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and computes y that satisfies (y/n)=a and $0<y<2^{k-2}$ of $\phi(x_{1,p},x_{1,q})$, $\phi(-x_{1,p},x_{1,q})$, $\phi(x_{1,p},-x_{1,q})$, and $\phi(-x_{1,p},-x_{1,q})$, where φ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem.

Furthermore, when

$$y = s||t \quad (s \in \{0,1\}^{k-k_0-2},\ t \in \{0,1\}^{k_0})$$

$$z = G(H(s) \odot t) \odot s,$$

the operation unit 204 is used to compute

$$m = \begin{cases} [z]^l & \text{if } [z]_{k_1} = 0^{k_1} \\ \text{“reject”} & \text{otherwise} \end{cases}$$

and by

the plaintext m is decrypted, where $[a]^k$ and $[a]_k$ denote first k-bits and last k-bits of a, respectively.

By using the above described method, for example, when d=3, it can be proved by equivalence with the difficulty of the problem of prime factorization of n that the public key encryption method is provably secure against adaptive chosen ciphertext attacks (proved for general trapdoor substitutions in the document 12).

According to the method of the present embodiment example, decryption processing is performed on a multiplication ring decided from a residue ring modulo pq, which is smaller than n, thereby achieving faster processing in comparison with conventional methods.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions p=2p'+1 and q=2q'+1, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system. Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

(Eighth embodiment example)

In this embodiment example, a, which is part of ciphertext in the seventh embodiment example, is used as a public key.

FIG. 1 shows the system configuration of this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p,q,β)

- $p,\ q$: prime integers, $\quad p \equiv 3 \pmod 4),\ q \equiv 3 \pmod 4)$
- $\beta \in \mathbb{Z}, \quad \alpha\beta \equiv 1 \ (\mathrm{mod}\ \mathrm{lcm}(p-1, q-1))$

satisfying

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,k_0,k_1,α,a,G,H) satisfying

- $n = p^d q$ $\quad (d > 1$ is odd)
- $k, k_0, k_1 \in \mathbb{Z}$: k is a binary length of pq, and k_0, k_1 are positive integers with $k > k_0 - k_1 - 2$.
- $\alpha \in \mathbb{Z}$
- $a \in \{-1, 1\}$
- $G : \{0,1\}^{k_0} \rightarrow \{0,1\}^{k-k_0-2}$
- $H : \{0,1\}^{k-k_0-2} \rightarrow \{0,1\}^{k_0}$

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Encryption and decryption processing

(1) The sender A selects a random number r($r \in \{0,1\}^{k0}$} for plaintext m ($m \in \{0,1\}^l$, $l=k-k_0-k_1-2$) by using the random number generating unit 101, uses the operation unit 103 within the sender device 100 to compute the following expression satisfying a=(x/n)

$$x = (m0^{k_1} \odot G(r)) || (r \odot H(m0^{k_1} \odot G(r)))$$

and further uses the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 to compute

$$C = x^{2n\alpha} \bmod n \;.$$

Furthermore, the sender A obtains the above public information from a third party or the receiver B and computes Jacobi's symbol a=(x/n) using the operation unit 103.

Furthermore, the sender A sends the ciphertext C to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

(2) The receiver B computes

$$
\begin{aligned}
x_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\
x_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q
\end{aligned}
$$

from the ciphertext C, using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and computes y that satisfies $(y/n)=a$ and $0<y<2^{k-2}$ of $\phi(x_{1,p},x_{1,q})$, $\phi(-x_{1,p},x_{1,q})$, $\phi(x_{1,p},-x_{1,q})$, and $\phi(-x_{1,p},-x_{1,q})$, where ϕ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem.

Furthermore, when

$$y = s||t \quad (s \in \{0,1\}^{k-k_0-2},\ t \in \{0,1\}^{k_0})$$

$$z = G(H(s) \odot t) \odot s,$$

the operation unit 204 is used to compute

$$m = \begin{cases} [z]^l & \text{if } [z]_{k_1} = 0^{k_1} \\ \text{“reject”} & \text{otherwise} \end{cases}$$

and by

the plaintext m is decrypted, where $[a]^k$ and $[a]_k$ denote first k-bits and last k-bits of a, respectively.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

Secret keys p and q can also be generated from expressions p=2p'+1 and q=2q'+1, where p' and q' are prime integers.

In the public key encryption method of the present embodiment example, the value of d (d>1) is changeable depending on a system.

Thereby, where the bit length of plaintext m is always small, decryption processing can be performed rapidly by increasing the range of d in a range in which prime factorization of n is intractable.

(Ninth embodiment)

In this embodiment example, a description will be made of how the storage medium with a computing function 400 which has poor computation capability such as an IC card computes ciphertext C, using the sender device 100 having high computation capability in the seventh and eighth embodiment examples. FIG. 8 outlines this embodiment example.

The storage medium with a computing function 400 generates plaintext m ($m \in \{0,1\}^l$, $l=k-k_0-k_1-2$), using the plaintext creating unit 406. Furthermore, the storage medium with a computing function 400 generates a random number r ($r \in \{0,1\}^{k0}$} using the random number

$$x = (m0^{k_1} \odot G(r)) \| (r \odot H(m0^{k_1} \odot G(r)))$$

generating unit 407 and uses the operation unit 403 to compute

from functions G and H. Furthermore, the storage medium with a computing function 400 computes

$$C_1 = x^{2\alpha} \bmod n$$

using the power computing unit 401 and the modulo calculation unit 402 from the public keys α and n, and outputs it to the input device 107 of the sender device 100 from the output device 405.

The sender device 100 uses the power-computing unit 102 and the modulo calculation unit 104 to compute ciphertext C by

$$C = {C_1}^n \bmod n$$

(Tenth embodiment)

In this embodiment, a description will be made of a public key encryption method which is a variant of the public key encryption methods of the first to fifth embodiment examples and the seventh and eighth embodiment examples, and is not provably secure but is excellent in the efficiency of encryption and decryption processing.

In the first to fifth embodiment examples, the operation unit 103 within the sender device 100 is used to compute the ciphertext C by

$$C = m^{2\alpha} \bmod n$$

In the first to fifth embodiment examples, the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200 are used to compute $m_{1,p}$ and $m_{1,q}$ from the ciphertext C by

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta}{4}} \bmod q \end{aligned}$$

In the seventh and eighth embodiment examples, the operation unit 103 within the sender device 100 is used to compute the ciphertext C by

$$C = x^{2\alpha} \bmod n$$

and in the seventh and eighth embodiment examples, the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200 are used to compute $m_{1,p}$ and $m_{1,q}$ from the ciphertext C by

$$\begin{aligned} m_{1,p} &= C^{\frac{(p+1)\beta}{4}} \bmod p, \\ m_{1,q} &= C^{\frac{(q+1)\beta}{4}} \bmod q \end{aligned}$$

(Eleventh embodiment)

In this embodiment, a description will be made of the case where identification information a is omitted in the seventh and eighth embodiments.

In this case, the sender A selects a random number $r(r\in\{0,1\}^{k0}\}$ for plaintext m (m∈{0,1}^{l}, l=k-k$_0$-k$_1$-2) by using the random number generating unit 101, uses the operation unit 103 within the sender device

$$x = (m0^{k_1} \odot G(r))||(r \odot H(m0^{k_1} \odot G(r))$$

100 to compute

and further uses the operation unit 103, the power computing unit 102, and the modulo calculation unit 104 within the sender device 100 to compute

$$C = x^{2n\alpha} \bmod n$$

Furthermore, the sender A sends the ciphertext C to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

The receiver B computes

$$\begin{aligned} x_{1,p} &= C^{\frac{(p+1)\beta q^{-1}}{4}} \bmod p, \\ x_{1,q} &= C^{\frac{(q+1)\beta p^{-d}}{4}} \bmod q \end{aligned}$$

from the ciphertext C, using the above described secret information (p,q,β) held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and for each of $y_1(x_{1,p},x_{1,q})$, $y_2(-x_{1,p},x_{1,q})$, $y_3(x_{1,p},-x_{1,q})$, and $y_4(-x_{1,p},-x_{1,q})$, when

$$y_i = s_i||t_i \quad (s_i \in \{0,1\}^{k-k_0-2},\ t_i \in \{0,1\}^{k_0},\ 1 \le i \le 4)$$

$$z_i = G(H(s_i) \odot t_i) \odot s_i \quad (1 \le i \le 4),$$

uses the operation unit 204 to compute

and decrypts the plaintext m by

$$m = \begin{cases} [z_i]^l & \text{if } [z_i]_{k_1} = 0^{k_1} \\ \text{“reject”} & \text{otherwise} \end{cases}$$

ϕ denotes ring isomorphism mapping from Z/(p)×Z/(q) to Z/(pq) by the Chinese remainder theorem. $[a]^k$ and $[a]_k$ denote first k-bits and last k-bits of a, respectively.

FIG. 10 shows comparisons between the method of the eleventh embodiment example and a typical practical public key encryption method in efficiency (the number of modular products) and security. In the comparisons in FIG. 10, α and β each are set equal to 1. Many of data in FIG. 10 are quoted from the document 9.

(Twelfth embodiment example)

In this embodiment example, a description will be made of a public key encryption method by which a public key encryption method described in the document 4 is subjected to a transformation method described in the document 12 to further increase the efficiency of decryption processing.

FIG. 1 shows the system configuration of this embodiment example. FIG. 9 outlines this embodiment example.

1. Key generation processing

The receiver B in advance generates secret information (p_i,β) $(1\leqq i\leqq h)$ satisfying

- p_i : prime integers $(p_i \equiv 3 \pmod 4),\ 1 \leq i \leq h)$
- $\beta \in \mathbb{Z},\quad \alpha\beta \equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$

by using the key generating unit 201 within the receiver device 200, generates public information (n,k,k_0,k_1,α,G,H) satisfying

- $n = \prod_{i=1}^{h} p_i$
- $k, k_0, k_1 \in \mathbb{Z}$: k is a binary length of n, and k_0, k_1 are positive integers with $k > k_0 - k_1 - 2$.
- $\alpha \in \mathbb{Z}$
- $G : \{0,1\}^{k_0} \rightarrow \{0,1\}^{k-k_0}$
- $H : \{0,1\}^{k-k_0} \rightarrow \{0,1\}^{k_0}$

and outputs the public information over the communication line 300 to send it to the sender device 100 or publicize it. The public information can be publicized using a known method such as, e.g., registration to a third party (public information managing institution). Other information is stored in the memory 205.

2. Encryption and decryption processing

The sender A selects a random number $r(r \in \{0,1\}^{k0}\}$ for plaintext m $(m \in \{0,1\}^{l}, l=k\text{-}k_0\text{-}k_1\text{-}2)$ by using the random number generating unit 101 within the sender device 100 to compute

$$x = (m0^{k_1} \odot G(r)) || (r \odot H(m0^{k_1} \odot G(r)))$$

and further obtains the above public information from a third party or the receiver B and uses the operation unit 103, the power computing unit 102, and the remainder computing unit 104 to compute

$$C = x^{2\alpha} \bmod n$$

Furthermore, the sender A sends the ciphertext C to the receiver device 200 of the receiver B over the communication line 300, using the communication device 106.

3. Decryption processing

$$x_i = C^{\frac{(p_i+1)\beta}{4}} \bmod p_i$$

The receiver B computes

from the ciphertext C, using the above described secret information (p_i, β) $(1 \leqq i \leqq h)$ held, and the power computing unit 202, the modulo calculation unit 203, and the operation unit 204 within the receiver device 200, and for 2^h pieces of $\{\phi(e_1x_1,e_2x_2,\ldots,e_hx_h) \mid e_1,\ldots,e_h\in\{-1,1\}\}$,

$$y_i = s_i||t_i \quad (s_i \in \{0,1\}^{k-k_0},\ t_i \in \{0,1\}^{k_0},\ 1 \leq i \leq 2^h)$$

when

$$z_i = G(H(s_i) \odot t_i) \odot s_i \quad (1 \leq i \leq 2^h)$$

uses the operation unit 204 to compute

$$m = \begin{cases} [z_i]^l & \text{if } [z_i]_{k_1} = 0^{k_1} \\ \text{“reject”} & \text{otherwise} \end{cases}.$$

and decrypts the plaintext m by

ϕ denotes ring isomorphism mapping from $Z/(p_1)\times Z/(p_2)\times\ldots\times Z/(p_h)$ to $Z/(n)$ by the Chinese remainder theorem. $[a]^k$ and $[a]_k$ denote first k-bits and last k-bits of a, respectively.

In the above described public key encryption method, with α and β each set equal to 1, by deleting α and β from public key and secret key respectively, key information in the method of the present embodiment example can be reduced.

By sending identification information such as the magnitudinous relationship of x and n/2, Jacobi's symbol (x/n) together with the ciphertext (or by creating x according to identification information specified by the public information), efficiency can be increased in decrypting of correct plaintext from 2^h pieces of $\{\phi(e_1x_1,e_2x_2,\ldots,e_hx_h) \mid e_1,\ldots,e_h\in\{-1,1\}\}$.

The method of this embodiment example solves the difficult problem of unique decryption, under the assumption that, with the conventional public key encryption method described in the document 4, security is provable in the case where n, which is part of public key, is the product of there or more mutually different prime integers.

Although the embodiment examples have been described in a general form that a sender and a receiver perform cipher communications using their respective devices, the present invention is actually applied to various systems.

For example, in an electronic shopping system, a sender is a user and a sender device is a computer such as a personal computer, while a receiver is a retail shop and a receiver device is a computer such as a personal computer. In this case, orders for user products and the like are often encrypted in common key cipher, and an encryption key used at that time is encrypted by the methods of the embodiment examples and sent to the device of the retail shop.

In an electronic mail system, respective devices are computers such as personal computers, sender's messages are often encrypted in common key cipher, and an encryption key used at that time is encrypted by the methods of the embodiment examples and sent to a receiver computer.

The present invention is applicable to other various systems in which conventional public key encryption methods are used.

Although computations in the embodiment examples are performed by the CPU executing programs within memory, besides by programs, data may be exchanged between a hard-wired computing unit and other computing units, and the CPU.

According to the present invention, there can be provided a public key encryption method and a key sharing method that are secure against chosen plaintext attacks, and the most powerful adaptive chosen ciphertext attacks, and enable high-speed processing, and devices and a system applying the methods.

The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the claims.